**AMENDMENTS TO THE SPECIFICATION:**

Please amend the specification as follows:

Replace the paragraph beginning at page 10, line 5 with the following paragraph:

This effect becomes more remarkable if the step of storing the priority includes the steps of: changing the priority of the each server so as to indicate the highest priority, when the state of the each server is changed to the single master state; changing the priority of the each server so as to indicate the second highest priority, when the state of the each server is changed to the master state; changing the priority of the each [[sever]] server so as to indicate the lowest priority, when the state of the each server is changed to the slave state; and prohibiting the priority of the each server from changing, when the state of the each server is changed to the halt state.

Replace the paragraph beginning at page 12, line 14 with the following paragraph:

The computer system shown in FIG. 1 has a structure that either the servers 100a or 100b carries out a process and even if a fault occurs in one server which is currently executing the process, the other server is capable of taking over the process. Particularly, a feature of this computer system is that if faults ~~occurs~~ occur in both of the servers 100a and 100b and, at least one of them is thereafter restored from the fault, this computer system is capable of determining accurately a server which should take over the process, that is, a server which carried out the process most recently. Such a determining system will be described later.

Replace the paragraph beginning at page 14, line 15 with the following paragraph:

(1) When two servers get into fault and after that, the two servers are restored from the fault, whether or not the process can be carried out by any one of the two servers is determined. If possible, a server which should carry out the process is determined so as to start the process on that server. The other server gets into a state which allows taking over of the process. The state which allows taking over of the process means sending information necessary for taking over the process from a server which is now carrying out the process to the other [[sever]] server so that even if a fault occurs in the server now carrying out the process, the other server is capable of taking over the process from that server.

Replace the paragraph beginning at page 18, line 9 with the following paragraph:

A state in which two servers are in fault at the same time and halted is indicated by (X X). When both of the servers are restored from the fault, as shown in FIG. 2, the state-transition, from 1-1-1 to 1-1-2, is carried out or the state-transition, from 1-2-1 to 1-2-2, is carried out. If both of the servers are restored from the fault, one of the ~~severs~~ servers starts the process. To secure a system condition which enables the taking over of the process, information necessary for the taking over is transmitted from the server carrying out the process to the other server (the state-transition 1-1-1 or 1-2-1). This is (SL SM) or (SM SL) state. If the system condition enabling the taking over of the process is secured after that, one server is kept in the slave state while the other server is changed from single master to master, that is, the state is changed to (SL M) or

(M SL) state (the state-transition 1-1-2 or 1-2-2). Even with this condition, the information necessary for taking over of the process is sent from the master to the slave.

Replace the paragraph beginning at page 20, line 21 with the following paragraph:

If a fault occurs in a server, the server is always halted (state X). This is a state-transition, i.e., any one of the transitions 5-1 to 5-8. Respective actions will be described below. If fault occurs in both of the servers with one in the [[slave]] master state and the other one in the slave state, namely (SL M) or (M SL), state-transitions of 5-1 to 5-5 are carried out so that the (X X) state is achieved. A fault may occur in one of the servers in the single master state, with information for taking over of the process being transmitted from the single master server to a server set in the slave state, in (SL SM) or (SM SL) state. In this case the process is impossible. Thus, both of the servers are halted that is, become in the (X X) state. This is state-transition of 5-2 or 5-6. If fault occurs in both of the servers with (SL SM) or (SM SL) state, the state-transition of 5-2 or 5-6 is carried out so that both of the servers are halted. If a fault occurs in a server in the slave state with (SL SM) or (SM SL) state, state-transition of 5-3 or 5-7 is carried out, so that the state of (X SM) or (SM X) is achieved. Under this state, the process is continued by a remaining server (server in single master state). Next, if with one server in single master and the other one halted, that is, with (X SM) or (SM X) state, a fault occurs in the server in single master state, both of the servers are halted, that is, (X X) is attained. This is state-transition of 5-4 or 5-8.